Unsupervised Behavior Extraction via Random Intent Priors

Hao Hu · MIG · Tsinghua

10/9/2023

Machine Intelligence Group

清華大學 交叉信息研究院
Tsinghua University Institute for Interdisciplinary Information Sciences

Motivation

- Abundant reward-free data, containing useful human beheaviors
- How to extract them effectively from offline data?

Random Neural Networks as Priors

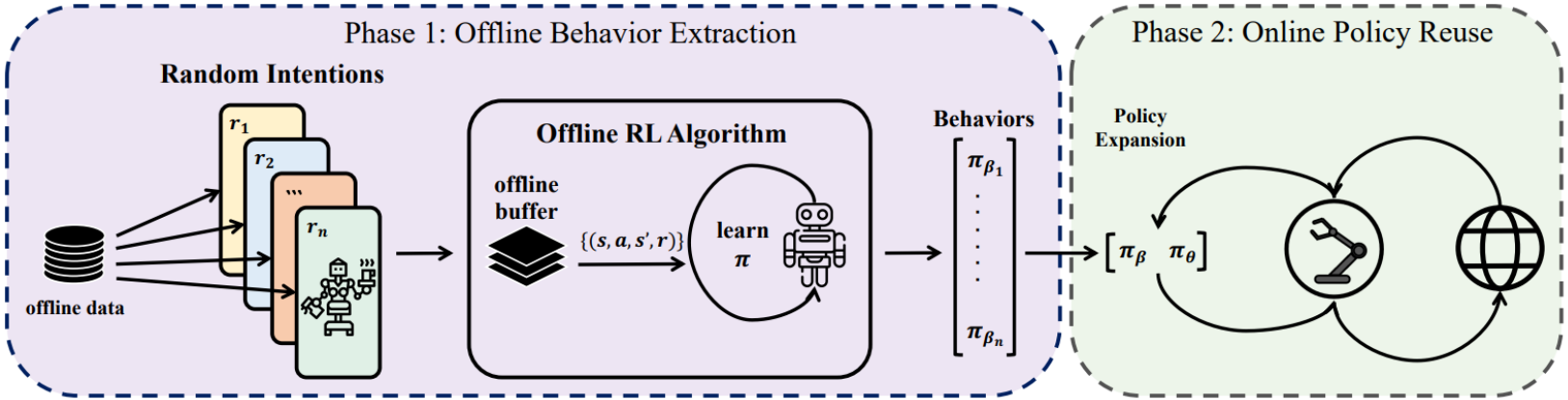


Figure 2: The framework of UBER. The procedure consists of two phases. In the first phase, we extract diverse and useful behaviors from the offline dataset with random rewards. In the second phase, we reuse previous behavior to accelerate online learning.

Policy Composition

- Policy set $\Pi = [\pi_\beta, \pi_\theta]$
- Utility $P_{\mathbf{w}}[i] = \frac{\exp(Q_\phi(s, a_i)/\alpha)}{\sum_j \exp(Q_\phi(s, a_j)/\alpha)}, \quad \forall i \in [1, \cdots K]$
- Composition

$$\tilde{\pi}(a|s) = [\delta_{a\sim\pi_\beta(s)}, \delta_{a\sim\pi_\theta(s)}]\mathbf{w}, \quad \mathbf{w} \sim P_{\mathbf{w}}$$

Algo: Random Neural Networks as Priors

Algorithm 1 Phase 1: Offline Behavior Extraction

Require: Behavior size N, offline reward-free dataset $\mathcal{D}^{\text{off}}$, prior intention distribution β
Initialize parameters of N independent offline agents $\{Q_{\theta_i}, \pi_{\phi_i}\}_{i=1}^{N}$
for $i = 1, \cdots, N$ **do**
    Sample a reward function from priors $z_i \sim \beta$
    Reannotate $\mathcal{D}^{\text{off}}$ as $\mathcal{D}_i^{\text{off}}$ with reward r_{z_i}
    for each training iteration **do**
        Sample a random minibatch $\{\tau_j\}_{j=1}^{B} \sim \mathcal{D}_i^{\text{off}}$
        Calculate $\mathcal{L}_{\text{critic}}^{\text{offline}}(\theta_i)$ and update θ_i
        Calculate $\mathcal{L}_{\text{actor}}^{\text{offline}}(\phi_i)$ and update ϕ_i
    end for
end for
Return $\{\pi_{\phi_i}\}_{i=1}^{N}$

Algo: Random Neural Networks as Priors

Algorithm 2 Phase 2: Online Policy Reuse

Require: $\{\pi_{\phi_i}\}_{i=1}^{N}$, offline dataset $\mathcal{D}^{\text{off}}$
Initialize online agents Q_θ, π_w and replay buffer $\mathcal{D}^{\text{on}}$
Construct expanded policy set $\widetilde{\pi} = [\pi_{\phi_1}, \ldots, \pi_{\phi_N}, \pi_w]$
for each iteration **do**
    Obtain initial state from environment s_1
    for step $t = 1, \cdots, T$ **do**
        Construct $P_{\widetilde{\pi}}$ According to Equation (6)
        Pick an policy to act $\pi_t \sim P_{\widetilde{\pi}}$, $a_t \sim \pi_t(\cdot|s_t)$
        Store transition (s_t, a_t, r_t, s_{t+1}) in $\mathcal{D}^{\text{on}}$
        for $i = 1, \cdots, N$ **do**
            Calculate $\mathcal{L}_{\text{critic}}^{\text{online}}(\theta)$ and update θ
        end for
        Calculate $\mathcal{L}_{\text{actor}}^{\text{online}}(w)$ and update w
    end for
end for

Theoretical Analysis: Completeness

Proposition A.1 (Proposition 4.1 restated)**.** *For any behavior π, there exists an intention z with reward function r_z such that π is the optimal policy under r_z, i.e.*

$$V^{\pi}_{h,r_z}(\cdot) = V^{*}_{h,r_z}(\cdot). \tag{9}$$

Moreover, if π is deterministic, then π is the unique optimal policy under r_z, in the sense that $\forall \pi^{*}_{r_z}, \pi^{*}_{r_z}(\cdot|s) = \pi(\cdot|s), \forall d^{\pi}(s) > 0.$

Proof. Let $r_z(s,a) = \mathbb{1}(d^{\pi}_{s_0}(s,a) > 0)$, then we have

$$V^{\pi}_{h,r_z}(\cdot) = V_{h,\max} = V^{*}_{h,r_z}(\cdot). \tag{11}$$

Theoretical Analysis: Completeness

Theorem A.2 (Theorem 4.2 restated.). *Consider linear MDP as defined in Definition 2.1. With an offline dataset $\mathcal{D}$ with size N, and the PEVI algorithm (Jin et al., 2021), the suboptimality of learning from an intention $z \in \mathcal{Z}$ with size $|\mathcal{Z}|$ satisfies*

$$\text{SubOpt}(\widehat{\pi}; r_z) \le 4c\sqrt{\frac{C_z^\dagger d^3 H^3 \iota}{N}}, \tag{12}$$

with probability $1-\delta$ for sufficiently large N, where $\iota = \log \frac{4dHK|\mathcal{Z}|}{\delta}$ is a logarithmic factor, c is an absolute constant and

$$C_z^\dagger = \max_{h\in[H]} \sup_{\|x\|=1} \frac{x^\top \Sigma_{\pi_z,h} x}{x^\top \Sigma_{\rho_h} x},$$

with

$$\Sigma_{\pi_z,h} = \mathbb{E}_{(s,a)\sim d_{\pi_z,h}(s,a)}[\phi(s,a)\phi(s,a)^\top], \quad \Sigma_{\rho_h} = \mathbb{E}_{\rho_h}[\phi(s,a)\phi(s,a)^\top].$$

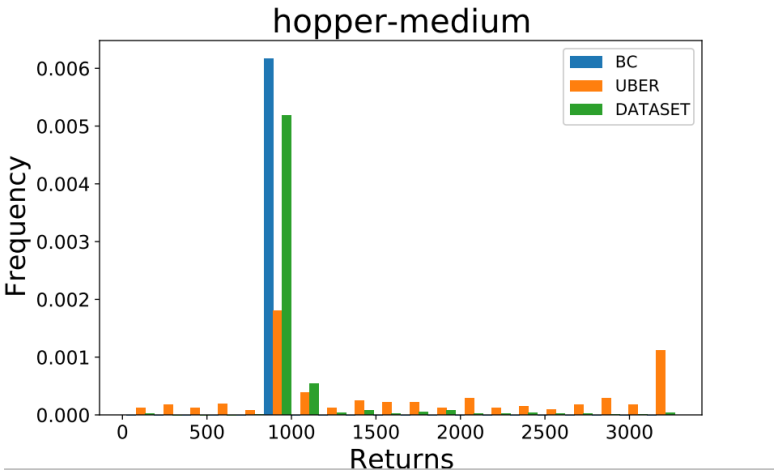
Theoretical Analysis: Coverage

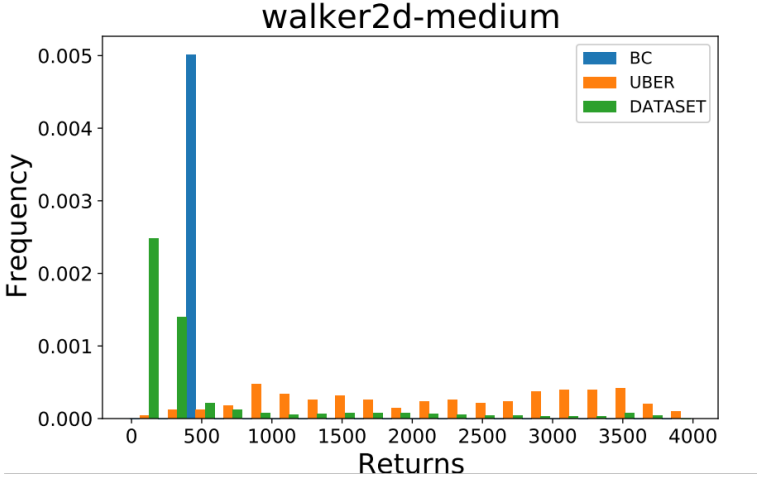
Theorem 4.3. *Assume the reward function admits a RKHS represention* $\psi(s,a)$ *with* $|\psi(s,a)| \leq \kappa$ *almost surely. Then with* $N = c_0\sqrt{M}\log(18\sqrt{M}\kappa^2/\delta)$ *random reward functions, the linear combination of the set of random reward functions can approximate the true reward function with error*

$$\epsilon \leq c_1 \log^2(18/\delta)/\sqrt{M},$$

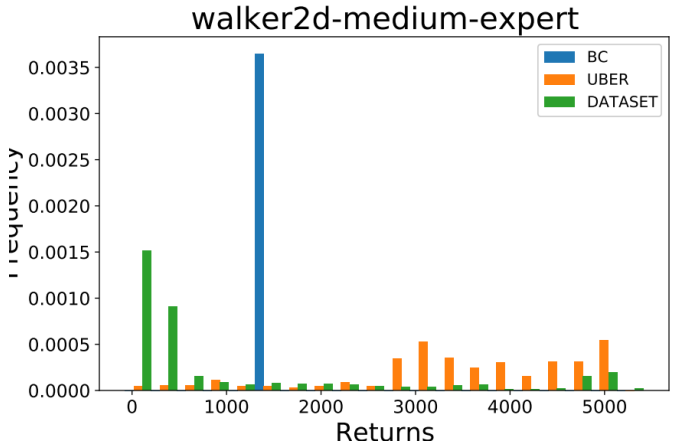
with probability $1-\delta$.

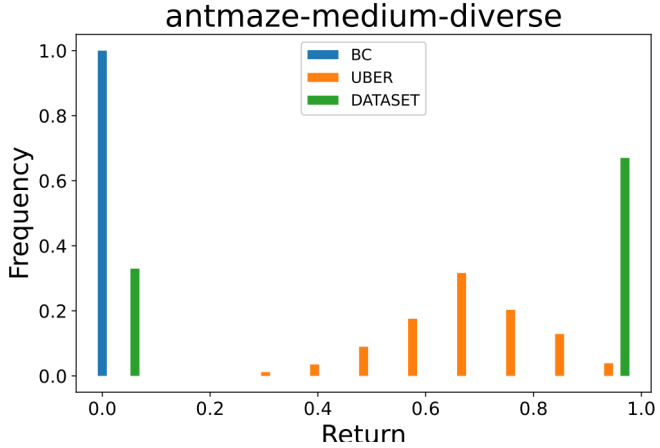
Experiments



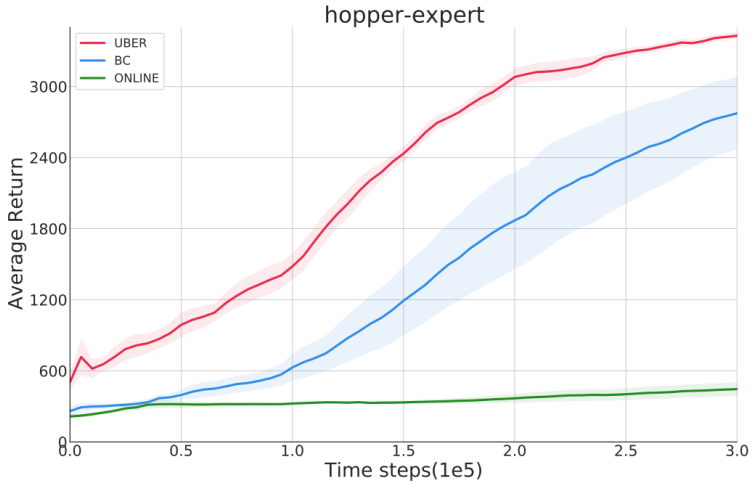


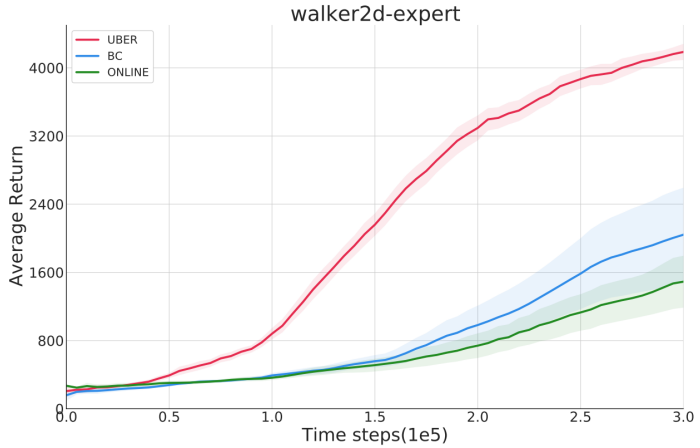
Experiments



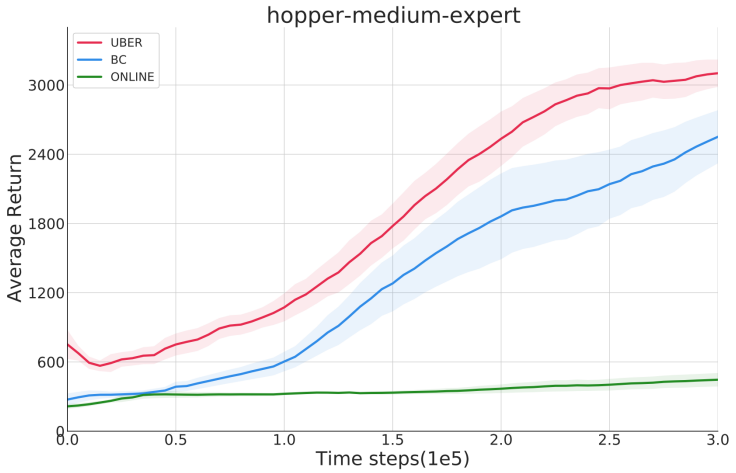


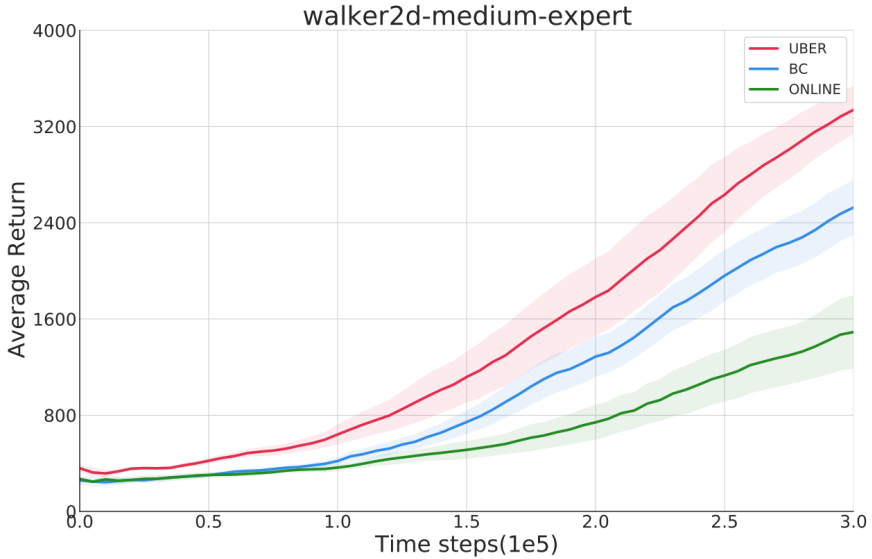
Experiments





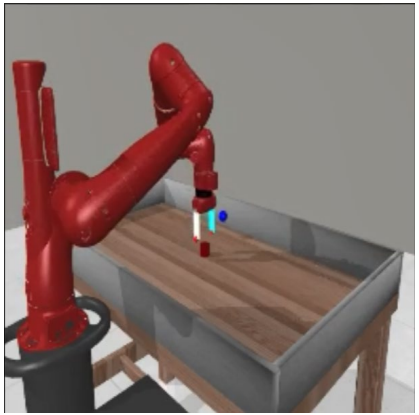
Experiments

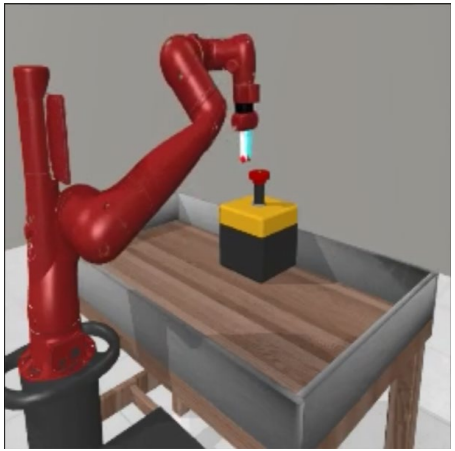




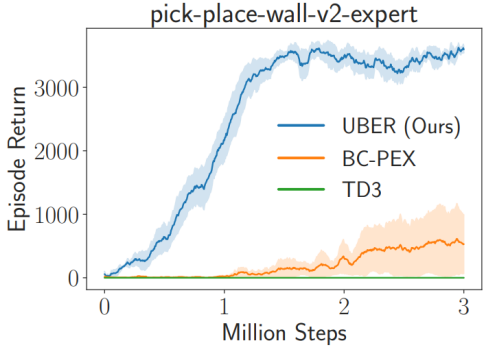
Multi-task: Meta-world

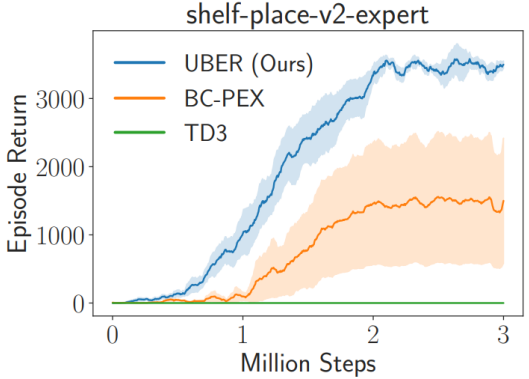
- Source: Push, Reach, Pick-place
- Target: Hammer, Peg-Insert-Side, Push-Wall, Pick-Place-Wall, Push-Back, Shelf-Place

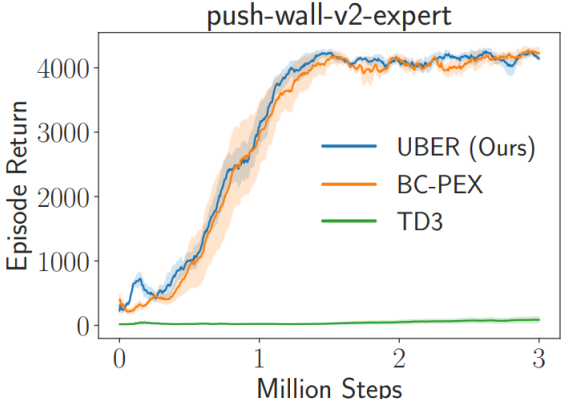


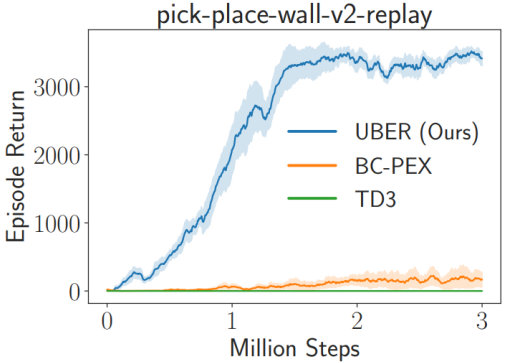


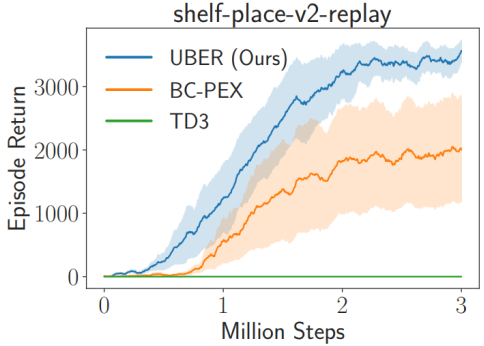
Results

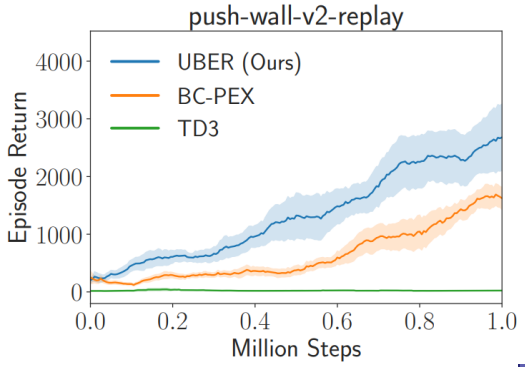

pick-place-wall-v2-expert
Episode Return
Million Steps
UBER (Ours)
BC-PEX
TD3


shelf-place-v2-expert
Episode Return
Million Steps
UBER (Ours)
BC-PEX
TD3

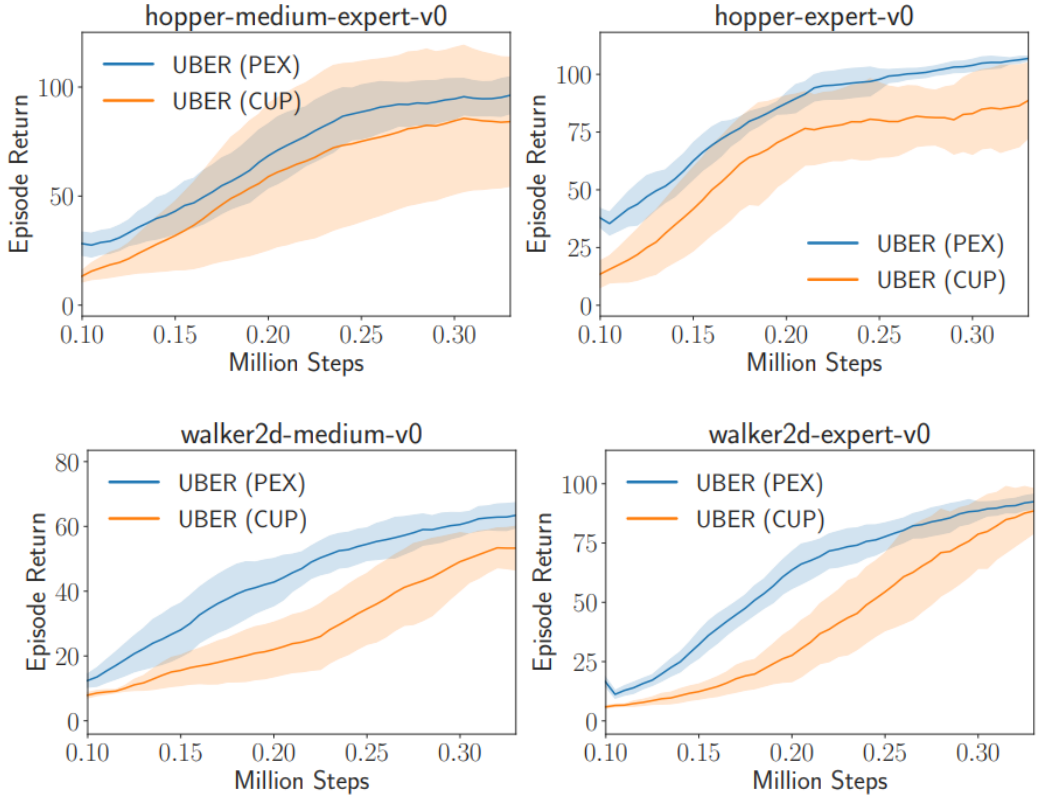

push-wall-v2-expert
Episode Return
Million Steps
UBER (Ours)
BC-PEX
TD3


pick-place-wall-v2-replay
Episode Return
Million Steps
UBER (Ours)
BC-PEX
TD3

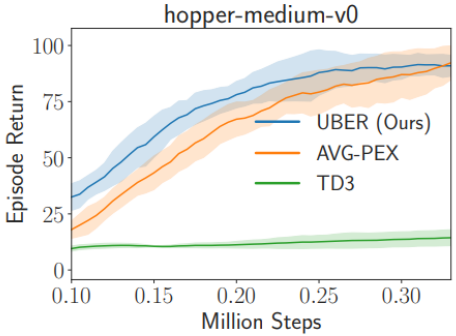

shelf-place-v2-replay
Episode Return
Million Steps
UBER (Ours)
BC-PEX
TD3

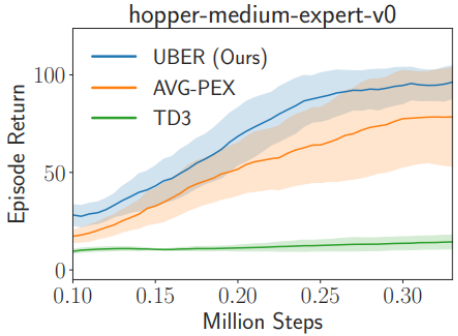

push-wall-v2-replay
Episode Return
Million Steps
UBER (Ours)
BC-PEX
TD3

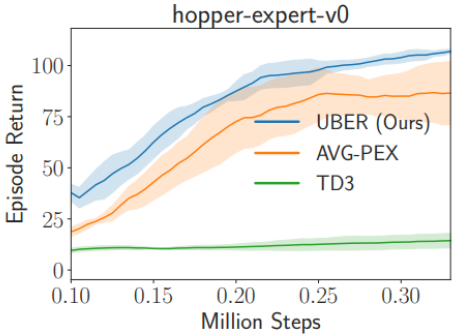
Ablations

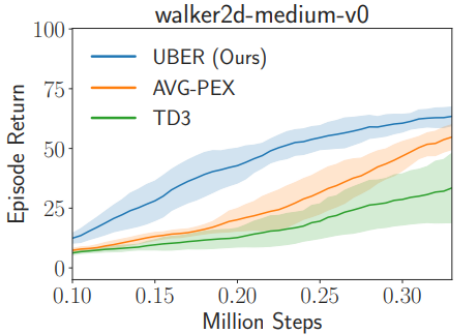

hopper-medium-expert-v0
UBER (PEX)
UBER (CUP)
Episode Return
Million Steps
hopper-expert-v0
UBER (PEX)
UBER (CUP)
Episode Return
Million Steps
walker2d-medium-v0
UBER (PEX)
UBER (CUP)
Episode Return
Million Steps
walker2d-expert-v0
UBER (PEX)
UBER (CUP)
Episode Return
Million Steps

Ablations


hopper-medium-v0
Episode Return
Million Steps
UBER (Ours)
AVG-PEX
TD3


hopper-medium-expert-v0
Episode Return
Million Steps
UBER (Ours)
AVG-PEX
TD3


hopper-expert-v0
Episode Return
Million Steps
UBER (Ours)
AVG-PEX
TD3


walker2d-medium-v0
Episode Return
Million Steps
UBER (Ours)
AVG-PEX
TD3

Thanks!